COMPANY
CONFIDENTIAL

Microsoft Word for the Apple Macintosh
Binary File Format Documentation

Information in this document is subject to change without notice and does not represent a commitment on the part of Microsoft Corporation. Copyright Microsoft Corporation 1984, 1985

COMPANY
CONFIDENTIAL

Revised 11/1/85

Microsoft Word Binary File Format
(on Apple Macintosh)

A Word binary file has three parts: header, text and formatting information. The formatting information is divided into character, paragraph, footnote, division and page table.

All fc (file character) fields are byte numbers stored as four bytes with highest order byte stored first. Bit 7 is the least significant bit; bit 0 is the most significant bit.

DEFINITIONS

sector	128 byte segment of file (bytes 0-127 are in sector 0, bytes 128-255 are in sector 1, etc.)
twip	1/20 point, 1/1440 inc
CHP	Character property
FOD	Character or paragraph run descriptor
FNTB	Footnote table
FND	Footnote descriptor
FPROP	File paragraph or character property
PAP	Paragraph property
PGD	Page descriptor
PGTB	Page number table
SED	Division descriptor
SEP	Division property
SETB	Division table
TBD	Tab descriptor

HEADER

The header contains "magic words" and pointers to the subdivisions of the formatting section as well as information about the length of the file:

Byte	*Meaning*	*Comment*
0..1	wIdent	Must be 0177062 octal
2..3	dty	Must be 0
4..5	wTool	Must be 0125400 octal
6..7	cReceipts	Reserved; must be 0
8..9	cbReceipts	Reserved; must be 0
10..11	bReceipts	Reserved; must be 0
12..13	isgMac	Reserved; must be 0
14..17	fcMac	Number of bytes of actual text PLUS 128 (bytes in one sector) [high byte first]
18..19	pnPara	Sector number of start of PARAGRAPH info
20..21	pnFntb	Sector number of FNTB (or pnSep if none)
22..23	pnSep	Sector number of SEP's (or pnSetb if none)
24..25	pnSetb	Sector number of SETB (or pnBftb if none)
26..27	pnBftb	Sector number of PGTB for Word document (or pnMac if none) (Sector number of Buffer table for glossary)
28..29	pnMac	Total sectors in file (last sector number +1)
32..97	szSsht	Reserved; must be 0
98..127	(unused)	Reserved; must be 0

TEXT

The text of the file starts at byte 128 (sector 1). It is ASCII text with the following restrictions:

- PARAGRAPH ENDS are stored as <Return> (ASCII 13 dec.). No other occurrences of this character are allowed.
- HARD LINE BREAKS which are not paragraph ends are stored as 11 dec.
- Other line breaks or WORD WRAP information is not stored.
- BREAKING HYPHENS are stored as ASCII 45 dec. (normal hyphen code); NON-REQUIRED HYPHENS are 31.
- NON-BREAKING SPACES are stored as 202. Normal SPACES are ASCII 32.
- FORM FEEDS are ASCII 12. (normal)
- TAB characters are ASCII 9. (normal)
- FOOTNOTE is ASCII 5 with character property SPECIAL on.
- (page) symbol used in page numbering is 1 with char property SPECIAL.
- DIVISION is ASCII 12 (form feed).

Note: If a paragraph and section end at the same "point in text," write a section mark (also called sections) not a paragraph mark in the text.

"Point in text" is any text in front of the footnote text:

File: HEADER
non footnote
text Page Boundary
Footnote text
Properties, etc.

Note: The text character at the end of the last footnote should be PARAGRAPH END (13) not a section mark (12). For that matter, all footnotes should have a PARAGRAPH MARK as their last character.

FORMATTING: CHARACTER AND PARAGRAPH

The CHARACTER section begins at the first complete sector after the end of the TEXT section (pnChar = (fcMac + 127) / 128).

The PARAGRAPH section begins at the first complete sector after the end of the character section and must be at pnPara as stored in the header. Both the character and paragraph sections are structured as a set of sectors. Each sector has the following format:

Byte	*Meaning*	*Comment*
0..3	fcFirst	Byte number in file of first character covered by this sector of formatting info. (fcFirst for the first character in the text is 128.)
4..	rgfod	An array of FODs and FPROPs (see below)
..126	grpfprop	
127	cfod	Number of FOD's in this sector

COMPANY
CONFIDENTIAL

FOD's are stored sequentially from byte 4 upward. FPROPs are downward from byte 126. Any unused space between them is undefined.

The structure of each FOD is as follows (these are fixed size):

Byte	*Meaning*	*Comment*
0..3	fcLim	Byte number in file AFTER last character covered by this FOD.
4..5	bfprop	Byte offset from beginning of sector - 4 to corresponding FPROP for these characters or this paragraph. FFFF means that there is no FPROP, i.e. no difference from standard character/paragraph.

Note: For last FOD in the last paragraph sector, fcLim =fcMac + 1 (fcMac <- in header record)

The structure of each FPROP is as follows: (these are variable size)

Byte	*Meaning*	*Comment*
0	cch	Number of bytes in this FPROP excluding this byte.
1..n	rgchProp	A prefix of a CHP (for characters) or a PAP (for paragraphs) sufficient to include all bits which differ from the standard CHP or PAP.

The format of a CHP is:

Byte	*Bit*	*Meaning*	*Comment*
0	0	fStyled	Reserved; must be 0
	1..7	stc	Reserved; must be 0
1	0	fBold	Characters are bold
	1	fItalic	Characters are italic
	2..7	ftc	Low five bits of font code (Standard is 2)
2		hps	Size of font in half pts (Standard is 24 dec.)
3	0	fUline	Characters are underlined
	1	fStrike	Characters are overstruck
	2	fDline	Characters are double underlined
	3	fUnused	
	4..5	csm	Case modifier, 0 normal, 1 upper, 3 small caps
	6	fSpecial	
	7	(unused)	Reserved; must be 0
4	0..2	ftcExtra	High 3 bits of font code
	3	fOutline	Characters are outlined
	4	fShadow	Characters are shadowed
	5..7	(unused)	
5		hpsPos	0 for normal; positive for superscript; negative for subscript
6..9		(unused)	Reserved; must be 0

COMPANY
CONFIDENTIAL

The standard CHP has byte 0 = 200 octal, byte 1 = 02 octal, byte 2 = 24 dec., all other bytes = 0. Note therefore that each character FPROP must have a cch of at least 1.

The format of a PAP is:

Byte	*Bit*	*Meaning*	*Comment*
0	0	fStyled	Reserved; must be 0
	1..7	stc	Reserved; must be 0
1	0..1	jc	Justification: 0=left, 1=center, 2=right, 3=both
	2	fKeep	Keep lines together
	3	fKeepFollow	Keep with following paragraph
	4..7		Reserved; must be 0
2	0..6	stcNormChp	Reserved; must be 0
	7	(unused)	Reserved; must be 0
3		(unused)	Reserved; must be 0
4..5		dxaRight	Right indent in twips
6..7		dxaLeft	Left indent in twips
8..9		dxaLeft1	First line left indent in twips (relative to dxaLeft)
10..11		dyaLine	Inter-line spacing in twips (Standard is 0)
12..13		dyaBefore	Spacing before paragraph in twips (standard is 0)
14..15		dyaAfter	Spacing After paragraph in twips (standard is 0)
16	0..3	rhc	Running head code, 1 bottom, 2 odd, 4 even 8 first
	4	fGraphics	Paragraph is a picture
	5..7	(unused)	Reserved; must be 0
17..21		(unused)	Reserved; must be 0
22..61		rgtbd[20]	Up to 20 TBDs

The format of a TBD is:

Byte	*Bit*	*Meaning*	*Comment*
0..1		dxa	indent from left margin of tab stop (in 20ths of a point)
2	0..2	jc	Justification of text AFTER tab: left=0, center=1, right=2, aligned=3 others reserved
	3..5	tlc	Leader code: white=0, dots=1, hyphens=2, underline=3, others reserved
	6..7	opcode	Operation code for Format Tabs
3		chAlign	ASCII code of character to align on if jcTab=3, or 0 to align on "."

The standard PAP has byte 0 = 200 octal, and all other bytes = 0. Note therefore that each paragraph FPROP must have cch >= 1.

COMPANY
CONFIDENTIAL

PAP's for picture paragraphs are standard PAP's with fGraphics = true. The text for a picture paragraph starts with six bytes describing position and scaling of the picture. They are:

Byte	*Meaning*
0..1	Distance from left margin in screen units
2..3	Horizontal increase in size in screen units
4..5	Vertical increase in size in screen units

In constructing the sectors of formatting information, there is a difference between paragraph and character FOD's. A character FOD may describe any number of consecutive characters with the same formatting. However, there MUST BE EXACTLY ONE PARAGRAPH FOD PER TEXT PARAGRAPH. In either case, it is allowable, and encouraged, to have multiple FOD's point to one FPROP in the same sector. No FOD may point to a different sector.

There must be no "holes" in either the character or paragraph formatting information; each must begin with the first text character (byte 128) and continue through the last. Therefore, the last character FOD and the last paragraph FOD must have fcLim = fcMac as defined in the header.

Important: The fcLim on the last paragraph FOD must be fcMac + 1.

FORMATTING: FOOTNOTES

The FOOTNOTE section [optional] begins at the first complete sector after the PARAGRAPH section and contains an FNTB which contains an array of FNDs.

The structure of the FNTB section [optional] is as follows:

Byte	*Meaning*	*Comment*
0..1	cfnd	Number of FND (1 or more)
2..3	cfndMax	(Same as above)
4..?	rgfnd	Array of FNDs plus 0 padding to fill sector

The structure of an FND is as follows:

Byte	*Meaning*	*Comment*
0..3	cpRef	Byte address of footnote reference from beginning of text (byte number in file - 128)
4..7	cpFtn	Byte address of footnote from beginning of text (byte number in file - 128)

NOTE: for last FND entry cpRef = fcMac +1 - 128 (fcMac <- in header record)
cpFtn = fcMac - 128

COMPANY
CONFIDENTIAL

The SEP section [optional] is in the sector immediately after the FOOTNOTE section and contains 1 or more SEPs.

The SETB section [optional] begins at the first complete sector after the end of the SEP section and must be at pnSetb as stored in the header. It contains an array of SEDs. Each SED contains an fcSep that points back to a SEP.

The format of a SEP [optional] is as follows:

Byte	*Bit*	*Meaning*	*Comment*
0		cch	Count of bytes used, EXCLUDING this byte. All properties at byte positions greater than cch will be set to their standard values.
1	0	fStyled	Reserved; must be 0
	1..7	stc	Reserved; must be 0
2	0..2	bkc	Break code: 0 line, 1 column, 2 page, 3 resto, 4 verso
	3..5	nfcPgn	Page number format code: 0 arabic, 1 upper case roman, 2 lower case roman, 3 upper case letter, 4 lower case letter
	6..7	(unused)	Reserved; must be 0
3..4		yaMac	Page length in twips(standard is 11*1440 = 15840)
5..6		xaMac	Page width in twips (standard is 8.5*1440 = 12240)
7..8		pgnStart	Starting page number (standard is -1)
9..10		yaTop	Top margin in twips (standard is 1440)
11..12		dyaText	Height of text in twips (standard is 9 * 1440 = 12960)
13..14		xaLeft	Left margin in twips (standard is 1.25 * 1440 = 1800)
15..16		dxaText	Width of text area in twips (standard is 6 * 1440 = 8640)
17	0..3	rhc	Running hd code, must be same as PAP
	4..5	(unused)	Reserved; must be 0
	6	fAutoPgn	Print pgns without hdr (standard is 0)
	7	fEndFtns	Footnotes at end of doc (standard is 0)
18		cColumns	number of columns (standard is 1)
19..20		yaRH1	Position of top header in twips (standard is 0.75 * 1440 = 1080)

COMPANY
CONFIDENTIAL

21..22	yaRH2	Position of bottom header in twips (standard is 10.25 * 1440 = 14760)
23..24	dxaColumns	Intercolumm gap in twips (standard is 0.5 * 1440 = 720)
25..26	dxaGutter	Gutter width in twips (standard is 0)
27..28	yaPgn	Y position of page number in twips (standard is 0.75 * 1440 = 1080)
29..30	xaPgn	X position of page number in twips (standard is 7.25 *1440 = 10440)
31..??	(unused)	Reserved; must be zero, padding to make same size as PAP

So we have yaTop + dyaText + (bottom margin, not stored) = yaMac, and xaLeft + dxaText + (right margin, not stored) = xaMac.

Note that if all of the above properties are standard, no SEP or SETB is needed at all. Otherwise, we have 1 <= cch <= 16.

The structure of the SETB section [optional] is as follows:

Byte	*Meaning*	*Comment*
0..1	csed	Number of SED (1 or more)
2..3	csedMax	Undefined, set to csed on doc load
4..?	rgsed	Array of SEDs plus 0 padding to fill sector (SEDs may be split across sectors)

The structure of a SED is as follows:

Byte	*Meaning*	*Comment*
0..3	cp	Byte address from start of text of first char of following division (byte number in file - 128)
4..5	fn	Undefined, value set on doc load
6..9	fcSep	Byte number in file of associated SEP.

Note: If file contains any footnotes, the cp for the *last* SED = (fcMac - 128)+ 1. Otherwise, cp = fcMac - 128.

Note that, if all of the SEP properties are standard and no SEP is necessary, then no SETB is necessary or possible and the pnSetb in the header should be equal to pnMac.

FORMATTING: PAGE TABLE

The PGTB section [optional] is in the sector immediately after the SEP section.

The structure of the PGTB is as follows:

Byte	*Meaning*	*Comment*
0..1	cpgd	Number of PGDs (1 or more)
2..3	cpgdMax	Undefined, set to cpgd on doc load
4..?	rgpgd	Array of PGDs plus 0 padding to fill sector

COMPANY
CONFIDENTIAL

The structure of a PGD is as follows:

Byte	*Meaning*	*Comment*
0..1	pgn	Page number in printed word document
2..5	cpMin	Byte address from start of text of first char in printed page (byte number in file - 128)